REMARKS

Claims 1-3, 14-16 and 27-29 are pending.

Claims 1-3, 14-16 and 27-29 continue to stand rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 5,539,907 (Srivastava).

In reply to the last Office Action, the applicants amended independent claims 1, 14 and 27 to make it clear that the term "first compiled code state" refers to machine-specific object code corresponding to a first processor instruction set and the term "second compiled code state" refers to machine-specific object code corresponding to a different processor instruction set. Specifically, each of those claims has been amended to expressly recite:

> the first compiled code state comprising *machine-specific object code corresponding to a first processor instruction set* and the second compiled code state comprising *machine-specific object code corresponding to a different processor instruction set.*

See, claims 1, 14 and 27 as amended above. Srivastava does not teach or suggest translating the compiled machine code of a computer program into a "second compiled code state" where that second compiled code state "compris[es] *machine-specific object code* corresponding to *a different processor instruction set*" as recited in each independent claim (emphasis added).

In responding to the applicants' prior reply, the Examiner continues to identify Srivastava's "single linked code module in the form of a machine independent register transfer language" as the claimed "second compiled code state." But the problem with that argument is that Srivastava's "single linked code module" is in the form of a "*machine independent register transfer language*." It is not "machine dependent," and it does not correspond to a "different processor instruction set," as claimed. Therefore, it is improper to read the claimed "second compiled code state" on Srivastava's "single linked code module."

The Examiner points to the mention of a "CPU architectures description 19" in column 4, lines 40-60 of Srivastava as evidence that Srivastava is performing a translation from one instruction set to a different instructions set. Specifically, Srivastava states "The specific architectures of target computer systems can be maintained in a CPU architectures description 19." But a careful reading of Srivastava shows that Srivastava does not contemplate *translating* a computer program compiled to run on one processor instruction set into compiled machine code for a *different* processor instruction set.

Rather, Srivastava describes a method that allows a developer to modify a computer program compiled to run on a particular processor so that certain performance monitoring code can be added that will allow the developer to monitor the performance of the computer program when it runs on the *same* computer processor. The title of Srivastava is "System for Monitoring Computer System Performance." The "Summary of the Invention" explains how the monitoring is performed:

> In accordance with the invention, a system for monitoring the performance of ***a computer system***, ***while executing a program/modifies the program prior to execution***. The modifications to the program include calls to user analysis routines. The user analysis routines collect performance data while the program is executing.
>
> The program is created as a collection of source code modules in the form of a high level language. Each of the source code modules is compiled into a corresponding object code module. The object code modules are translated into a single linked code module in the form of a machine independent register transfer language. . . .
>
> The linked code module is partitioned into basic program components. . . .
>
> Fundamental instrumentation routines are provided to identify and locate specific program components, the identified specific program components are modified to call user analysis routines. The user analysis routines are combined with the modified linked code module. The linked code module is converted to machine executable code to be executed in ***the computer system*** so that performance data can be collected by the user analysis routines ***when the modified program is executed***.

(col. 2, ll. 29-63) (emphasis added). Thus, according to Srivastava, the object code modules of a computer program designed to run on a particular computer processor or architecture are (i) translated to an intermediate form (the "single linked code module in the form of a *machine independent* register transfer language"), (ii) monitoring code is added to the program in that intermediate form, and then (iii) a code generator generates new object code including the monitoring code for execution on the ***same*** computer processor. There is no translation of the original object modules into object code executable on a different processor.

The reference in Srivastava to different "CPU architectures" of target computer systems is not a reference to translating from one architecture to a different architecture. Rather, it simply means that Srivastava's monitoring process can be used on different processor architectures. But each time it is used, the system merely takes the object code modules of a computer program designed to run on a particular computer architecture, translates the object code to an intermediate form (the "single linked code module"), adds monitoring code to the program in that intermediate form, and then generates new object code including the monitoring code for execution on the ***same*** computer architecture. There is no mention or suggestion in Srivastava that the new object code would be translated to a different computer architecture. That would be contrary to the purpose of Srivastava's monitoring system, which is to monitor the performance of a computer program on a particular computer system.

Because the "single linked code module" of Srivastava is not "machine-specific" and does not correspond to a "different processor instruction set" as claimed, and because Srivastava does not otherwise suggest translation from one processor instruction set to another, the applicants respectfully submit that Srivastava does not teach or suggest the invention recited in independent claims 1, 14 and 27. Inasmuch as claims 2-3, 15-16 and 28-29 depend either directly or indirectly from one of those independent claims, the applicants submit that they too are patentable for the same reasons. Withdrawal of the Section 102(b) rejection of claims 1-3, 14-16 and 27-29 is respectfully requested for this reason.

CONCLUSION

For all the foregoing reasons, the applicants respectfully submit that the present application is in condition for allowance. An early Notice of Allowance is respectfully requested.

Date: August 1, 2008

/Steven B. Samuels/
Steven B. Samuels
Registration No. 37,711

Woodcock Washburn LLP
Cira Centre
2929 Arch Street, 12th Floor
Philadelphia, PA 19104-2891
Telephone: (215) 568-3100
Facsimile: (215) 568-3439

Correspondence address:
Richard J. Gregson, Esq.
Unisys Corporation
Unisys Way
Blue Bell, Pennsylvania 19424-0001